N70-23099

NASA TECHNICAL MEMORANDUM

NASA TM X-1983

NASA TM X-1983

A COMPUTER PROGRAM FOR RAY TRACE THROUGH SPACECRAFT WINDOWS

by Kenneth C. White

Ames Research Center
Moffett Field, Calif.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • APRIL 1970

1. Report No. NASA TM X-1983	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle A COMPUTER PROGRAM FOR RAY TRACE THROUGH SPACECRAFT WINDOWS		5. Report Date April 1970
		6. Performing Organization Code
7. Author(s) Kenneth C. White		8. Performing Organization Report No. A-3456
9. Performing Organization Name and Address NASA Ames Research Center Moffett Field, Calif., 94035		10. Work Unit No. 125-17-02-13-00-21
		11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D. C., 20546		13. Type of Report and Period Covered Technical Memorandum
		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A FORTRAN IV computer program has been written specifically for tracing rays through spacecraft windows. The program computes the window-induced angular deviation of the rays from their original path. Any type of window may be considered with no restriction on size, shape, material, or number of panes. The necessary equations are written in vector matrix form for mathematical convenience and ease in computation. The program requires mathematical models of each window surface. The numerical procedures used are described and a test case is presented.

17. Key Words Suggested by Author(s)	18. Distribution Statement
Ray tracing Spacecraft windows Computer program	Unclassified - Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	33	$3.00

*For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151

SYMBOLS

Program Symbol	Equation Symbol	Definition
A	A	$(Z_1 - \Delta I_1 \gamma)\gamma$
ALPHAI	α_i	azimuth angle of initial incident ray
ALPHAR	α_r	azimuth angle of final refracted ray
BETAI	β_i	angle initial I makes with y-axis
CI(I), I=1,3	I_x, I_y, I_z	components of $\hat{I}$ in x, y, z directions
CN(I), I=1,3	N_x, N_y, N_z	components of $\hat{N}$ in x, y, z directions
CR(I), I=1,3	R_x, R_y, R_z	components of $\hat{R}$ in x, y, z directions
CROSSR	$\lvert\hat{I}\times\hat{R}\rvert$	magnitude of vector product $\hat{I}\times\hat{R}$
D(I), I=1,N		distances from RP1 to other RP's
DELALP	$\alpha_i - \alpha_r$	out-of-plane deviation
DELDEL	$\delta_i - \delta_r$	in-plane deviation
DELINC		angle between initial $\hat{I}$ and $\hat{R}$
DELTAA	ΔI_2	$\Delta I_1 + A$
DELTAI	δ_i	elevation angle of $\hat{I}$
DELTAP(I), I=1,N		scale factor for interfaces between surfaces
DELTAR	δ_r	elevation angle of $\hat{R}$
DELZ	Δz	window deformation at a given point (x, y)
DOTN	$\hat{I}\cdot\hat{R}$	dot product of $\hat{I}$ and $\hat{R}$
DOTP	$\hat{I}\cdot\hat{N}$	dot product of $\hat{I}$ and $\hat{N}$
DUM		M(XV) in ITER
DUMN1		M(XVN) in NORMAL

Program Symbol	Equation Symbol	Definition
DUMN2		M(XV) in NORMAL
E(I), I=1,3	E_x, E_y, E_z	components of $\overline{E}$ in x, y, z directions
FI		index used in NORMAL
GAMMAI	γ_i	angle between initial $\hat{I}$ and z-axis
I,J,K,L		indices
MA		number of rows or columns in matrix A in XPYXM
MAGN	$1\overline{\nabla}F1$	magnitude of gradient of F
MB,MC		number of rows or columns in matrix B in XPYXM
N		number of window surfaces
NA,NB,NC		indices computed and used for matrix multiplication in XPYXM
QRI	$\frac{n_{k+1}}{n_k}$	ratio of refractive indices
RAD		conversion factor degrees to radians
RI(I), I=1,N+1	n_i	refractive indices of mediums
ROOT		$\left(\frac{n_1}{n_2}\right)^2 - 1.0 + (\hat{I}\cdot\hat{N})^2$
ROUT		$\sqrt{\text{ROOT}}$
SEC		conversion factor radians to arcsec
SIGMAI	σ_i	angle between initial $\hat{I}$ and x-axis
SM(I,J,K)	M_i	matrices for mathematical models of surface shapes
X	x	X coordinate
XN	$N_x\lvert\overline{\nabla}F\rvert$	component of $\overline{N}$ in x direction
XV	$\overline{X}$	column vector $\overline{X}$
XVN	$\frac{\partial\overline{X}}{\partial X}$	column vector $\frac{\partial\overline{X}}{\partial x}$

Program Symbol	Equation Symbol	Definition
Y	Y	Y coordinate
YN	$N_Y\lvert\overline{\nabla}F\rvert$	component of $\overline{N}$ in y direction
YV	$\overline{Y}$	column vector $\overline{Y}$
YVN	$\frac{\partial\overline{Y}}{\partial y}$	column vector $\frac{\partial\overline{Y}}{\partial y}$
Z	z	Z coordinate

A COMPUTER PROGRAM FOR RAY TRACE

THROUGH SPACECRAFT WINDOWS

Kenneth C. White

Ames Research Center

SUMMARY

A FORTRAN IV computer program has been written specifically for tracing rays through spacecraft windows. The program computes the window-induced angular deviation of the rays from their original path. Any type of window may be considered with no restriction on size, shape, material, or number of panes. The necessary equations are written in vector matrix form for mathematical convenience and ease in computation. The program requires mathematical models of each window surface. The numerical procedures used are described and a test case is presented.

INTRODUCTION

On-board navigation systems that derive their basic inputs from optical measurements made through a spacecraft window have been considered for manned space flight. The optical measurements of interest are the angle between two celestial bodies, such as a star and a planet, and the angle between a star and a spacecraft. These angular navigation measurements are subject to window-induced errors that result from the angular deviation or bending of the lines of sight as they pass through the window. In this report the terms "ray" and "line of sight" are used interchangeably. It should be noted that ray tracing is done in the opposite direction of the normal ray tracing.

The purpose of this program is to compute the angular line-of-sight deviations induced by the spacecraft windows. This is accomplished by tracing the lines of sight through the window by geometric ray tracing techniques. Many ray trace programs are available but they are primarily oriented toward lens design and usually require rotationally symmetric optical systems. A FORTRAN IV computer program has therefore been written which emphasizes the computation of the angular deviations of the rays and is not limited to rotationally symmetric optical systems.

The input to the program consists of mathematical models of each window surface, the orientation parameters of the lines of sight to be traced through the window, and the various parameters of the window environment. The output consists of the orientation parameters of the line of sight after it has passed through the window and the angular differences between the original line of sight and the deviated line of sight.

The basic equations are presented and the program is described fully in this report. This description includes the program listing, program usage, flow charts, and a sample case.

An application of the program and a discussion of analytical methods which utilize the program are given in reference 1.

PROGRAM DESCRIPTION

The presence of a spacecraft window in the path of rays of light causes these rays to bend or to deviate from their original path. This program is designed to determine the magnitude and direction of the angular deviations of the light rays. In order to determine the deviations, the ray must be traced through the window and a comparison made between the entering and exiting rays. The fundamental law of ray tracing is Snell's law of refraction,

$$n_1 \sin \theta_1 = n_2 \sin \theta_2$$

where

n_1 the index of refraction of medium 1

θ_1 the angle the incident ray makes with the normal to the interface between mediums 1 and 2

n_2 the index of refraction of medium 2

θ_2 the angle the refracted ray in medium 2 makes with the normal

Snell's law is used to trace a ray through several mediums by successive applications of the law at each interface between mediums such as air and glass. This is accomplished by letting the refracted ray become the incident ray for the next application of the law. In order to apply Snell's law, the normals to the interfaces between mediums at the points where the ray intersects the interface must be known. This means the shape of the interface must be known. An iteration scheme is also required to determine the points of intersection of the ray with the interfaces.

BASIC EQUATIONS

For mathematical convenience and ease in computation, Snell's law is expressed in vector form as follows:

$$\hat{R} = \left(\frac{n_1}{n_2}\right) \hat{I} + \left\{\sqrt{1 - \left(\frac{n_1}{n_2}\right)^2 \left[1 - (\hat{I} \cdot \hat{N})^2\right]} - \left(\frac{n_1}{n_2}\right) (\hat{I} \cdot \hat{N})\right\} \hat{N} \tag{1}$$

where $\hat{I}$ and $\hat{R}$ are the unit vectors in the direction of the incident and refracted rays. The unit vector $\hat{N}$ is in the direction of the normal to the interface between mediums 1 and 2 at the point of intersection of the incident ray; n_1 and n_2 are as previously defined. Equation (1) is derived in reference 2.

Coordinate System

The coordinate system is illustrated in figure 1. The innermost surface of the spacecraft window is assumed to lie in the xy-plane. The positive z-axis is toward the outside of the spacecraft. The incident and refracted rays and their orientation angles are also illustrated in the figure. The vector $\hat{I}$ is the unit vector in the direction of the incident ray. The azimuth angle, α_i, is defined as the angle in the xy-plane between the positive x-axis and the projection, $\hat{I}_p$, of $\hat{I}$ onto the xy-plane. The elevation angle, δ_i, is the angle between $\hat{I}$ and $\hat{I}_p$. The incidence angle, θ, is the complement of δ_i. The unit vector $\hat{R}$ is in the direction of the refracted ray which emerges from the outer-most window surface. The azimuth angle, α_r, is the angle in the xy-plane between the positive x-axis and the projection, $\hat{R}_p$, of $\hat{R}$ onto the xy-plane. The elevation angle, δ_r, is the angle between $\hat{R}$ and $\hat{R}_p$. The azimuth angles α_i and α_r are measured from the positive x-axis toward the positive y-axis and vary from 0° to 360°. The elevation angles, δ_i and δ_r, are measured from the projected vectors $\hat{I}_p$ and $\hat{R}_p$ toward the positive z-axis and vary from 0° to 90°.

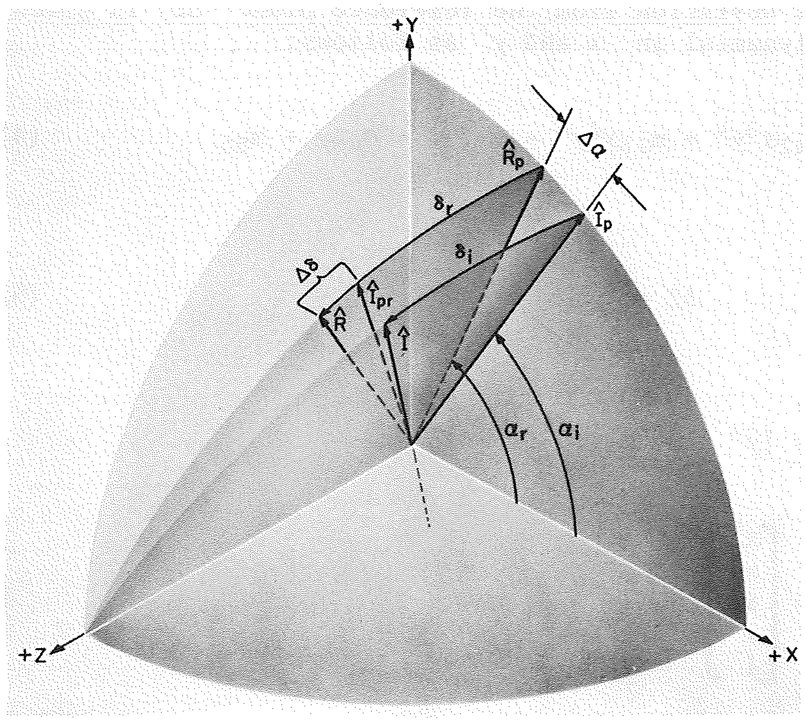


Figure 1.- Ray trace coordinate system.

The unit vectors $\hat{I}$ and $\hat{R}$ are given by the following equations:

$$\hat{I} = \cos\delta_i \cos\alpha_i \, \hat{i} + \cos\delta_i \sin\alpha_i \, \hat{j} + \sin\delta_i \, \hat{k} \tag{2}$$

$$\hat{R} = \cos\delta_r \cos\alpha_r \, \hat{i} + \cos\delta_r \sin\alpha_r \, \hat{j} + \sin\delta_r \, \hat{k} \tag{3}$$

where $\hat{i}$, $\hat{j}$, and $\hat{k}$ are unit vectors in the directions of the x, y, and z axes, respectively.

Two angular ray deviations $\Delta\alpha$ and $\Delta\delta$ are also defined in figure 1. The out-of-plane deviation is defined as $\Delta\alpha = (\alpha_i - \alpha_r)$. The in-plane deviation is defined as $\Delta\delta = (\delta_i - \delta_r)$. These angular deviations are the important results of the ray trace.

Window Surface Mathematical Models

The window surface shapes are described in terms of their deviations from reference planes set parallel to the xy-plane. The first reference plane is coincident with the xy-plane. There is a reference plane for each window surface. The window surface deviation from the reference plane ΔZ is given by a fourth degree mixed polynomial in x and y as follows:

$$\Delta Z = m_{11}x^4y^4 + m_{12}x^3y^4 + m_{13}x^2y^4 + \ldots + m_{45}y + m_{55} \tag{4}$$

or in shorthand form:

$$\Delta Z = \overline{Y}^T M\overline{X} \tag{5}$$

where

$$\overline{Y}^T = [y^4 \quad y^3 \quad y^2 \quad y \quad 1]$$

$$\overline{X} = \begin{bmatrix} x^4 \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix}$$

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} \\ m_{31} & m_{32} & m_{33} & m_{34} & m_{35} \\ m_{41} & m_{42} & m_{43} & m_{44} & m_{45} \\ m_{51} & m_{52} & m_{53} & m_{54} & m_{55} \end{bmatrix}$$

The elements of the matrix M are program inputs. The normal to the surface at a given point is obtained by evaluating the gradient of the surface at that point. If equation (5) is written in the form

$$F(x, y, z) = \Delta Z - \overline{Y}^T M\overline{X} \tag{6}$$

then the unit vector normal to the surface F is given by

$$N = \frac{\overline{\nabla}F}{|\overline{\nabla}F|} = \frac{\frac{\partial F}{\partial x}\hat{i} + \frac{\partial F}{\partial y}\hat{j} + \frac{\partial F}{\partial z}\hat{k}}{\sqrt{\left(\frac{\partial F}{\partial x}\right)^2 + \left(\frac{\partial F}{\partial y}\right)^2 + \left(\frac{\partial F}{\partial z}\right)^2}} \tag{7}$$

where

$$\left.\begin{aligned}
\frac{\partial F}{\partial x} &= -\overline{Y}^T M \frac{\partial \overline{X}}{\partial x} \\
\frac{\partial F}{\partial y} &= -\frac{\partial \overline{Y}^T}{\partial y} M\overline{X} \\
\frac{\partial F}{\partial z} &= 1 \\
\frac{\partial \overline{X}}{\partial x} &= \begin{bmatrix} 4x^3 \\ 3x^2 \\ 2x \\ 1 \\ 0 \end{bmatrix} \\
\frac{\partial \overline{Y}^T}{\partial y} &= [4y^3 \quad 3y^2 \quad 2y \quad 1 \quad 0]
\end{aligned}\right\} \tag{8}$$

Equations (4) through (8) are derived in reference 2.

Equations (1) through (8) together with the iteration scheme described in appendix A comprise the necessary elements to compute a ray trace.

Description of a Ray Trace

A block diagram illustrating the logical sequencing of the ray trace program is given in figure 2. The orientation angles of the incident ray, the coordinates of the reference point, the mathematical models of the window surfaces, and the window and environmental parameters are input. The components of the unit vector $\hat{I}$ in the direction of the incident ray are computed by means of equation (2). The components of the vector $\overline{E}$ from the origin to the reference point are computed. Next the point of intersection of $\hat{I}$ with the first reference plane is computed using equations in appendix A. Then the point of intersection of $\hat{I}$ with the first window surface is calculated using the iteration scheme described in appendix A. The iteration is done in subroutine ITER. After the point of intersection is found, the unit vector $\hat{N}$

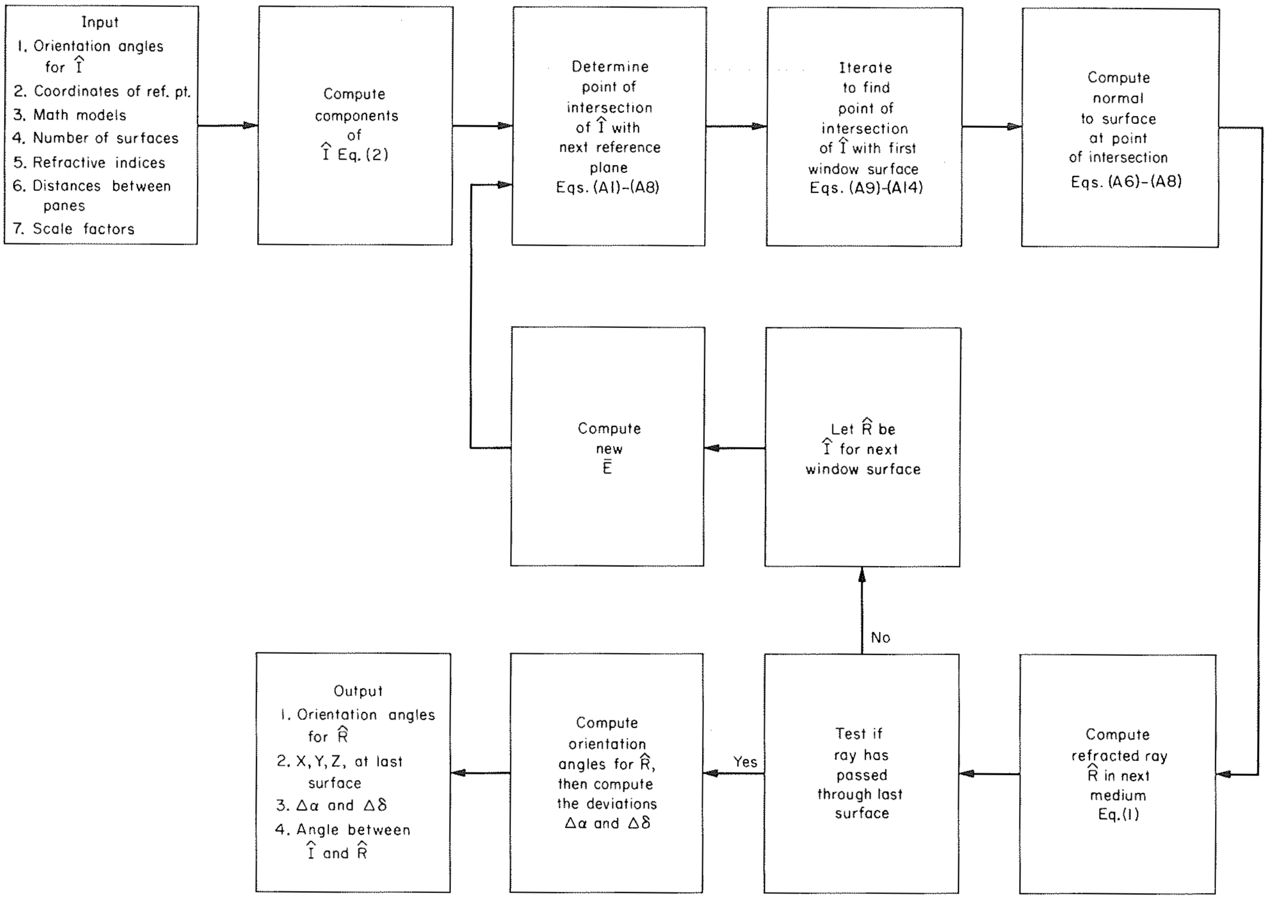


Figure 2.- Computer program schematic.

normal to the window surface at the point is computed by means of equations (6), (7), and (8). This is accomplished in subroutine NORMAL. The refracted ray in the next medium is computed by equation (1). This is done in subroutine REFRC. The number of surfaces traversed by the ray are tested. If the ray has passed through the last surface, the refracted ray is compared with the original incident ray to determine the angular deviations. If the ray has not passed through all the surfaces, the refracted ray becomes the incident ray and the ray trace is continued until the ray has completely passed through the window.

ROUTINE DESCRIPTIONS

The main program is designated MS2500 and is the main processing unit for the ray trace. Equation (2) and equations (A1) through (A8) are utilized in the main program. Input-output is, of course, accomplished here.

Subroutine ITER (MS2501) is used to compute the points of intersection of the ray with the window surfaces. Equations (4) and (5) and (A9) through (A14) are used in this subroutine. The calling statement is CALL ITER (X, Y, SM, CI, K, DELZ, DELTAP) where:

X x coordinate

Y y coordinate

SM SM(I, J, K) = M_i matrices for math models

CI CI(I) I = 1, 3 = I_x, I_y, I_z components of $\hat{I}$

K index

DELZ ΔZ

DELTAP scale factor

SM and CI are arrays. X, Y, SM, CI, K, and DELTAP are input from the main program. DELZ is computed in ITER and returned. New values of X and Y are computed also and returned.

Subroutine NORMAL (MS2502) is used to compute the normals to the window surfaces at the points of intersection. Equations (6), (7), and (8) are used in this subroutine. The calling statement is CALL NORMAL (X, Y, SM, CN, K, DELTAP) where:

X x coordinate

Y y coordinate

SM SM(I, J, K) = M_i matrices for the math models

CN CN(I) = $\hat{N}_x$, $\hat{N}_y$, $\hat{N}_z$ components of $\hat{N}$

K index

DELTAP scale factor

SM and CN are arrays. The components CN(I) of $\hat{N}$ are computed in the subroutine and returned to main program. Others are input from the main program.

Subroutine REFRC (MS2303) uses equation (1) to compute the refracted ray in each medium through which the ray passes. The calling statement is CALL REFRC (CI, CN, QRI, CR) where

CI CI(I) = I_x, I_y, I_z components of $\hat{I}$

CN CN(I) = N_x, N_y, N_z components of $\hat{N}$

QRI $\frac{n_{k+1}}{n_k}$ ratio of refractive indices

CR $CR(I) = R_x, R_y, R_z$ components of $\hat{R}$

CI, CN, and CR are arrays. The components CR(I) of the refracted ray $\hat{R}$ are computed in this routine and the other parameters come from the main program.

Subroutine XPYXM (MS2504) is a matrix multiplying subroutine used in subroutines ITER and NORMAL. The calling sequence is (A, B, C, NRA, NCA, NRB, NCB, J) where A and B are the matrices to be multiplied and NRA, NCA, NRB, and NCB are, respectively, the number of rows in A, columns in A, rows in B, and columns in B; C is the result of the matrix multiplication. The index J determines the multiplication to be performed. For

$$J = 1 \quad C = AB$$

$$J = 2 \quad C = AB^T$$

$$J = 3 \quad C = A^T B$$

Listings and flow charts for the main program and subroutines appear in appendixes B and C.

PROGRAM USAGE

The program is written in the FORTRAN IV computer language. It operates at Ames Research Center on an IBM 7094 computer under the IBJOB Processor of the IBSYS operating system, version 13. The program requires 2752_8 or 1514_{10} storage locations exclusive of the FORTRAN system subroutines required.

DECK MAKEUP

$JOB	CARD	
$IBJOB	CARD	
MS2300	DECK	Main program
MS2301	DECK	ITER
MS2302	DECK	NORMAL
MS2303	DECK	REFRC
MS2304	DECK	XPYXM
$DATA	CARD	
INPUT	CARDS	

PROGRAM INPUTS

Inputs to the ray trace program can be divided into three categories: constants, math model matrices, and individual ray data. The constants describe the window configuration and its environment. The math model matrices describe the window surface shapes. The individual ray data describe the rays to be traced through the window. The data are all input in card form. The input cards are described below.

Constants

Columns		Symbol	Definition
Card 1	Format I10		
1-10		N	Number of window surfaces
Card 2	Format 8E10.0		
1-10		D(1)	Distances from reference planes to other reference planes
11-20		D(2)	
21-30		D(3)	
31-40		D(4)	
41-50		D(5)	
Card 3	Format 8E10.0		
1-10		DELTAP(1)	Scale factors at interfaces between air and glass
11-20		DELTAP(2)	
21-30		DELTAP(3)	
31-40		DELTAP(4)	
41-50		DELTAP(5)	
51-60		DELTAP(6)	
Card 4	Format 8E10.0		
1-10		RI(1)	Refractive indices of each medium through which ray is traced
11-20		RI(2)	
21-30		RI(3)	
31-40		RI(4)	
41-50		RI(5)	
51-60		RI(6)	
61-70		RI(7)	

See figure 3.

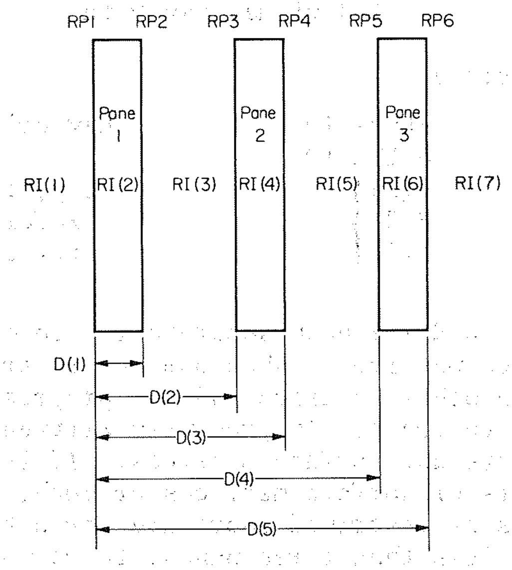


Figure 3.- Definition of reference planes, refractive indices, and input distances.

Math Model Matrices

Cards 5-9 Columns of math model matrix 1

Format 5E15.8 $J = 1, 5$

Columns	1-15	$M_{1,j}$
	16-30	$M_{2,j}$
	31-45	$M_{3,j}$
	46-60	$M_{4,j}$
	61-75	$M_{5,j}$

Cards 10-14 Columns of math model matrix 2

if N = 2 Same format

Cards 15-19 Columns of math model matrix 3

if N = 3 Same format

Cards 20-24 Columns of math model matrix 4

if N = 4 Same format

Cards 25-29 Columns of math model matrix 5

if N = 5 Same format

Cards 30-34 Columns of math model matrix 6

if N = 6 Same format

Individual Ray Data

Cards 35-∞ Format 8E10.0

1-10	ALPHA I	Ray orientation angles α_i, δ_i
11-20	DELTA I	
21-30	E(1)	x, y, z coordinates of reference position from which ray originates
31-40	E(2)	
41-50	E(3)	

There will be one card for each separate ray to be traced for a given set of constants and surface matrices. When one ray is traced through the window, the program looks for another to trace and the program is terminated when there are no more rays to trace. As presently written, the program considers only one set of constants and surface matrices. If it is desired to consider another set of constants or surface matrices or both, the program must be rerun. The input cards as described above are for a three-pane window. If the window has more or less than three panes, there would be a corresponding increase or decrease in the number of cards.

PROGRAM OUTPUTS

All the program inputs are printed out in the same sequence that they are input. In addition, for each input ray the following quantities are output:

Symbol	
ALPHAR	$\left.\begin{matrix}\alpha_r \\ \delta_r\end{matrix}\right\}$ orientation angles of final refracted ray on outside of window
DELTAR	
DELALP	$\alpha_i - \alpha_r = \Delta\alpha$
DELDEL	$\delta_i - \delta_r = \Delta\delta$
DELINC	Angle between incident ray and final refracted ray
$\left.\begin{matrix}x \\ y \\ z\end{matrix}\right\}$	x_f, y_f, z_f coordinates of point where refracted ray leaves outermost window surface

A sample data printout is included with this writeup.

The program requires 0.2 to 0.3 second of computation time to complete a single trace through a three-pane window.

TAPES

Logical tape 5 is used for input and logical tape 6 for output.

SAMPLE CASE DESCRIPTION

The printout for a sample ray trace follows this section. In the sample case the window consisted of one pane of glass 0.3 inch thick with a pressure of 5 psi imposed on it from within the spacecraft. The window is circular with a 7-inch diameter. Two rays, both with $\alpha_i = 0°$ and $\delta_i = 60°$, were traced through the window. One ray intersected the inner window surface at point X = 0, Y = 0 and the other at point X = 6, Y = 0.

SAMPLE CASE OUTPUT

```
CONSTANTS

N=  2
D=    0.30000000E 00 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38
DELTAP=    0.50000000E 01  0.50000000E 01 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38
RI=   0.10001023E 01  0.14580000E 01  0.10000000E 01 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38

MATRIX 1

 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38  0.65431000E-06

 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38

 -0.00000000E-38 -0.00000000E-38  0.13086200E-05 -0.00000000E-38 -0.16754500E-03

 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38

  0.65431000E-06 -0.00000000E-38 -0.16754500E-03 -0.00000000E-38  0.66386300E-02
MATRIX 2

 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38  0.65431000E-06

 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38

 -0.00000000E-38 -0.00000000E-38  0.13086200E-05 -0.00000000E-38 -0.16754500E-03

 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38 -0.00000000E-38

  0.65431000E-06 -0.00000000E-38 -0.16754500E-03 -0.00000000E-38  0.66386300E-02

INPUT DATA

 ALPHAI=  0.00000000E-38 DELTAI=  0.60000000E 02  X=   0.60000000E 01  Y=   0.00000000E-38  Z=  0.00000000E-38
OUTPUT DATA

ALPHAR=  0.00000000E-38 DELTAR=  0.59997527E 02 DELALP=  0.00000000E-38 DELDEL=  0.89011025E 01 DELINC=  0.89087864E 01

X=  0.61143256E 01  Y= -0.00000000E-38  Z=  0.30644726E 00


INPUT DATA

 ALPHAI=  0.00000000E-38 DELTAI=  0.60000000E 02  X=  0.00000000E-38  Y=  0.00000000E-38  Z=  0.00000000E-38
OUTPUT DATA

ALPHAR=  0.00000000E-38 DELTAR=  0.60002730E 02 DELALP=  0.00000000E-38 DELDEL= -0.98323988E 01 DELINC=  0.98254832E 01

X=  0.12869728E 00  Y= -0.00000000E-38  Z=  0.33317927E 00
```

CONCLUDING REMARKS

A FORTRAN IV computer program has been written for tracing rays computationally through spacecraft windows.

Program equations are written in vector-matrix form for ease in computation. The program requires 0.2 to 0.3 second to trace a ray through a three-pane window. The program determines the angular deviations of the ray as it passes through the window.

Rays may be traced through any window whose surface shapes can be described mathematically in polynomial form.

The ray trace program has been used extensively to trace rays through Gemini spacecraft optical windows as discussed in reference 1. The program is currently being used to trace rays through Apollo windows and also through generalized windows of various sizes and shapes.

Ames Research Center
National Aeronautics and Space Administration
Moffett Field, Calif., 94035, Dec. 5, 1969

APPENDIX A

COMPUTATION OF THE POINT OF INTERSECTION OF A RAY WITH THE REFERENCE PLANE AND THE WINDOW SURFACE

This appendix describes the determination of the intersection of the assumed interior ray $\hat{I}$ with the first reference plane RP1 and also the determination of the intersection of $\hat{I}$ with the first window surface. This is an iterative procedure, and is illustrated in figure 4. Here the window coordinate system is given as $(\hat{i}\hat{j}\hat{k})$; the vector from the origin locating either the eye of the observer, or some reference point on the instrument is given as $\bar{E}$, some initial ray vector as $\bar{I}_1$, which terminates in the x-y plane, and a vector in the x-y plane closing with $\bar{I}_1$ as $\bar{d}$. Let

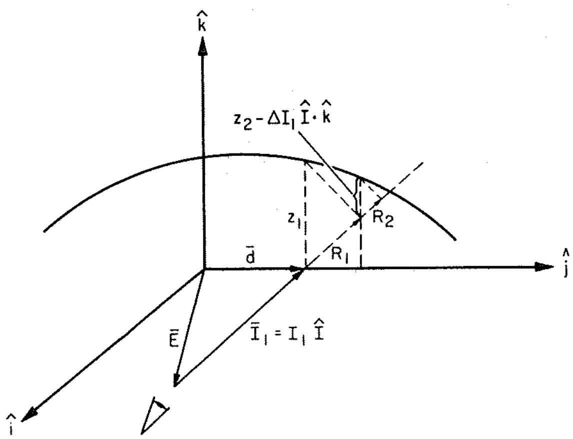


Figure 4.- Geometry of the scheme for iterating from the reference plane to the window surface.

$$\bar{E} = E_x\hat{i} + E_y\hat{j} + E_z\hat{k} \tag{A1}$$

$$\hat{I}_1 = \sigma\hat{i} + \beta\hat{j} + \gamma\hat{k} \tag{A2}$$

$$\bar{I}_1 = I_1\hat{I} \tag{A3}$$

where I_1 is unknown. Now $\bar{d}$ is of the form

$$\bar{d} = x_1\hat{i} + y_1\hat{j} + 0\hat{k} \tag{A4}$$

and from figure 4

$$\bar{d} = \bar{E} + I_1\hat{I} \tag{A5}$$

where

$$\left.\begin{aligned} x_1 &= E_x + I_1\sigma \\ y_1 &= E_y + I_1\beta \\ 0 &= E_z + I_1\gamma \end{aligned}\right\} \tag{A6}$$

Then from the third equation

$$I_1 = -\frac{E_z}{\gamma} \tag{A7}$$

and from the first two

$$\left.\begin{aligned} x_1 &= E_x - \frac{\sigma}{\gamma} E_z \\ y_1 &= E_y - \frac{\beta}{\gamma} E_z \end{aligned}\right\} \tag{A8}$$

Now, if we set

$$\Delta I_1 = 0 \tag{A9}$$

compute

$$z_1 = \Delta p \overline{Y}_1^T [M] \overline{X}_1 \tag{A10}$$

and project the increment of z_1 above the $\overline{I}_1$ vector onto the $\hat{I}$ direction

$$A = (z_1 - \Delta I_1 \hat{I} \cdot \hat{k})\hat{k} \cdot \hat{I} = (z_1 - \Delta I_1 \gamma)\gamma \tag{A11}$$

Now form

$$\Delta I_2 = \Delta I_1 + A \tag{A12}$$

and compute

$$x_2 = x_1 + A\hat{I} \cdot \hat{i} = x_1 + A\sigma \tag{A13}$$

$$y_2 = y_1 + A\hat{I} \cdot \hat{j} = y_1 + A\beta \tag{A14}$$

These values of x_2 and y_2 are then used in the equation for z (eq. (A10)), and the computation is continued until $|A| < \varepsilon$. At present $\varepsilon = 1.0 \times 10^{-6}$ in the program. At this time the last values computed for x, and y, and z are the coordinates of the point of intersection of our ray with the refracting surface, to the accuracy we have specified for ε. The basic scheme of this iteration is that by successively projecting the value of $(z - A\hat{I} \cdot \hat{k})$ onto the vector $\hat{I}$, we approach the point of intersection with the window surface. This system appears to be stable for all continuous smooth surface functions.

APPENDIX B

PROGRAM LISTING

```
C       K.C.WHITE  GENERAL RAY TRACE THRU SPACECRAFT WINDOW
        DIMENSION E(3),CI(3),SM(5,5,6),DELTAP(6),CN(3),RI(7),CR(3),D(5)
C       INPUT DATA FOLLOWS
        READ(5,311)N                                                       1
  311   FORMAT(I10)
        READ(5,312)(D(I),I=1,5)                                            3
        READ(5,312)(DELTAP(I),I=1,6)                                      10
        READ(5,312)(RI(I),I=1,7)                                          17
  312   FORMAT(8E10.5)
        READ(5,315)(((SM(I,J,K),I=1,5),J=1,5),K=1,N)                      24
  315   FORMAT(5E15.8)
C        NOW PRINT INPUT DATA
        WRITE(6,316)N,(D(I),I=1,5),(DELTAP(I),I=1,6),(RI(I),I=1,7)        37
  316   FORMAT(1H0,9HCONSTANTS//1H ,3HN= ,I2/1H ,3HD= ,5E16.8/1H ,8HDELTAP
      1= ,6E16.8/1H ,4HRI= ,7E16.8//)
        DO 318 K=1,N
  318   WRITE(6,317)K,((SM(I,J,K),J=1,5),I=1,5)                           59
  317   FORMAT(1H ,7HMATRIX ,I1//(1H ,5E16.8)//)
  300   READ(5,308)ALPHAI,DELTAI,(E(I),I=1,3)                             70
  308   FORMAT(5E10.0)
        WRITE(6,309)ALPHAI,DELTAI,(E(I),I=1,3)                            77
  309   FORMAT(1H0,10HINPUT DATA//1H ,8H ALPHAI=,E16.8,8H DELTAI=,E16.8,4H
      1   X=,E16.8,4H   Y=,E16.8,4H   Z=,E16.8//)
        WRITE(6,307)                                                      84
  307   FORMAT(1H ,11HOUTPUT DATA//)
        K=1
        Z=0.0
        RAD=1.74532925E-02
        SEC=2.062648064E05
        IF(DELTAI.NE.90.0)GO TO 15
C       NOW COMPUTE COMPONENTS OF INCIDENT RAY
   17   CI(1)=0.0
        CI(2)=0.0
        CI(3)=1.0
        ALPHAI=ALPHAI*RAD
        DELTAI=DELTAI*RAD
        GO TO 16
   15   ALPHAI=ALPHAI*RAD
        DELTAI=DELTAI*RAD
        CI(1)=COS(DELTAI)*COS(ALPHAI)                                     92    93
        CI(2)=COS(DELTAI)*SIN(ALPHAI)                                     94    95
```

```
        MS2500     -  EFN    SOURCE STATEMENT  -  IFN(S)  -

         CI(3)=SIN(DELTAI)
C        NEXT COMPUTE POINT OF INTERSECTION OF INCIDENT RAY WITH XY PLANE      96
      16 SIGMAI=(ACOS (CI(1)))                                                 98
         BETAI=(ACOS (CI(2)))                                                  99
         GAMMAI=(0.15707963268E01-DELTAI)
      20 X= E(1)-E(3)*CI(1)/CI(3)
         Y=E(2)-E(3)*CI(2)/CI(3)
C        SUBROUTINE ITER COMPUTES INTERSECTION OF RAY WITH WINDOW SURFACE
         CALL ITER (X,Y,SM,CI,K,DELZ,DELTAP)                                   101
         Z=Z+DELZ
C        SUBROUTINE NORMAL COMPUTES NORMAL TO WINDOW SURFACE
         CALL NORMAL(X,Y,SM,CN,K,DELTAP)                                       103
         QRI=RI(K+1)/RI(K)
C        SUBROUTINE REFRC COMPUTES THE REFRACTED RAY IN THE NEXT MEDIUM
         CALL REFRC  (CI,CN,QRI,CR)                                            107
         IF (N-K) 500,500,40
      40 E(1)=X
         E(2)=Y
         E(3)=Z-D(K)
         DO 50 I=1,3
      50 CI(I)=CR(I)
         Z=D(K)
         K=K+1
         GO TO 20
C        AFTER TRACE COMPLETE ANGULAR RAY DEVIATIONS ARE COMPUTED
     500 CROSSR=SQRT((COS(BETAI)*CR(3)-COS(GAMMAI)*CR(2))**2+(COS(GAMMAI)*C
        1R(1)-COS(SIGMAI)*CR(3))**2+(COS(SIGMAI)*CR(2)-COS(BETAI)*CR(1))**2
        2)                                                                     123  124  125  126  127  128
                                                                               129
         DELINC=ARSIN(CROSSR)*SEC                                              130
         DELTAR=ARCOS(SQRT(CR(1)**2+CR(2)**2))                                 131  132
         ALPHAR=(ATAN2(CR(2),CR(1)))                                           133
         IF(ALPHAI)530,530,505
     505 IF(ALPHAR)520,530,530
     520 ALPHAR=ALPHAR+0.6283185072E01
     530 DELDEL= (DELTAI-DELTAR)*SEC
         DELALP=  ALPHAI-ALPHAR
         DELALP=DELALP*SEC
         DELTAR=DELTAR/RAD
         ALPHAR=ALPHAR/RAD
C        OUTPUT DATA FOLLOWS
         WRITE(6,510)ALPHAR,DELTAR,DELALP,DELDEL,DELINC                        139

                                                              06/12/69
        MS2500     -  EFN    SOURCE STATEMENT  -  IFN(S)  -

     510 FORMAT(1H0,7HALPHAR=,E16.8,8H DELTAR=,E16.8,8H DELALP=,E16.8,8H DE
        1LDEL=,E16.8,8H CELINC=,E16.8//)
         WRITE(6,501)X,Y,Z                                                     140
     501 FORMAT(1H0,2HX=,E16.8,4H  Y=,E16.8,4H  Z=,E16.8////)
         GO TO 300
         END
```

```
      MS2501     -  EFN   SOURCE STATEMENT  -  IFN(S)  -

      SUBROUTINE ITER (X,Y,SM,CI,K,DELZ,DELTAP)
C     SUBROUTINE ITER COMPUTES INTERSECTION OF RAY WITH WINDOW SURFACE
      DIMENSION SM(5,5,6),XV(5),YV(5),CI(3),DUM(5),DELTAP(7)
      J=1
      DELTAA=0.0
      XV(5)=1.0
      YV(5)=1.0
   80 DO 85 I=1,4
      L=5-I
      XV(L)=XV(L+1)*X
   85 YV(L)=YV(L+1)*X
C     SUBROUTINE XPYXM IS A VECTOR MATRIX MULTIPLY ROUTINE
      CALL XPYXM(SM(1,1,K),XV,DUM,5,5,5,1,1)                        16
      CALL XPYXM (YV,DUM,DELZ,5,1,5,1,3)                            18
      DELZ=DELZ*DELTAP(K)
      A=(DELZ-DELTAA*CI(3))*CI(3)
      IF(ABS(A)-1.0E-06)101,101,90
   90 DELTAA=DELTAA+A
      X=X+A*CI(1)
      Y=Y+A*CI(2)
      J=J+1
      IF(J-25)80,101,101
  101 RETURN
      END

      MS2502     -  EFN   SOURCE STATEMENT  -  IFN(S)  -

      SUBROUTINE NORMAL(X,Y,SM,CN,K,DELTAP)
C     SUBROUTINE NORMAL COMPUTES NORMAL TO WINDOW SURFACE
      DIMENSION SM(5,5,6),XV(5),YV(5),CN(3),DELTAP(6),XVN(5),YVN(5),
     1DUMN1(5),DUMN2(5)
      REAL MAGN
      XV(5)=1.0
      YV(5)=1.0
      XVN(5)=0.0
      YVN(5)=0.0
   80 DO 85 I=1,4
      L=5-I
      XV(L)=XV(L+1)*X
      YV(L)=YV(L+1)*X
      FI=I
      XVN(L)=FI*XV(L+1)
   85 YVN(L)=FI*YV(L+1)
C     SUBROUTINE XPYXM IS A VECTOR MATRIX MULTIPLY ROUTINE
      CALL XPYXM(SM(1,1,K),XVN,DUMN1,5,5,5,1,1)                     19
      CALL XPYXM (YV,DUMN1,XN,5,1,5,1,3)                            21
      CALL XPYXM(SM(1,1,K),XV,DUMN2,5,5,5,1,1)                      24
      CALL XPYXM (YVN,DUMN2,YN,5,1,5,1,3)                           26
      MAGN=SQRT ((XN**2+YN**2)*(DELTAP(K)**2)+1.0)                  30
      CN(1)=(-DELTAP(K)*XN)/MAGN
      CN(2)=(-DELTAP(K)*YN)/MAGN
      CN(3)=1.0/MAGN
      RETURN
      END
```

```
        MS2503      -  EFN   SOURCE STATEMENT  -  IFN(S)  -

      SUBROUTINE REFRC  (CI,CN,QRI,CR)
C     SUBROUTINE REFRC COMPUTES THE REFRACTED RAY IN THE NEXT MEDIUM
      DIMENSION CI(3),CN(3),CR(3)
      DOTP=0.0
      DO 110 I=1,3
  110 DOTP=DOTP+CI(I)*CN(I)
      ROOT=QRI**2-1.0+DOTP**2
      IF(ROOT)111,112,112
  111 ROUT=0.0
      WRITE(6,114)ROOT                                                 12
  114 FORMAT(1H0,6HROOT= ,E16.8/)
      GO TO 113
  112 ROUT=SQRT (QRI**2-1.0+DOTP**2)                                   15
  113 DO 120 I=1,3
  120 CR(I)=(CI(I)+(ROUT-DOTP)*CN(I))/QRI
      DOTN=CR(1)*CN(1)+CR(2)*CN(2)+CR(3)*CN(3)
      RETURN
      END
```

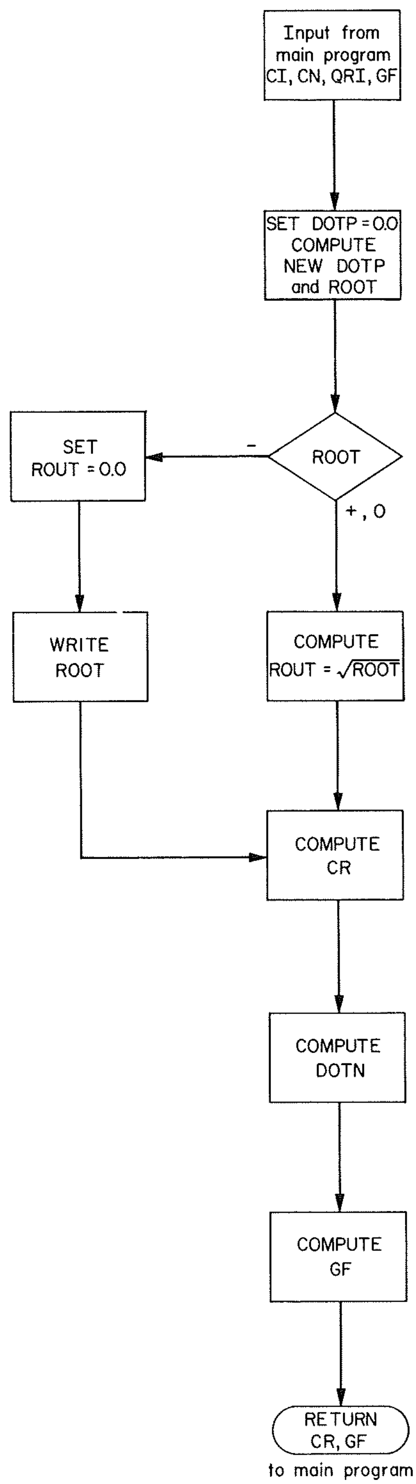
```
        MS2504      -  EFN   SOURCE STATEMENT  -  IFN(S)  -

      SUBROUTINE XPYXM(A,B,C,NRA,NCA,NRB,NCB,J)
C     SUBROUTINE XPYXM IS A VECTOR MATRIX MULTIPLY ROUTINE
      DIMENSION A(36),B(36),C(36)
      GO TO (13,14,15),J
   13 MA=NRA
      MB=NCB
      MC=NRB
      GO TO 16
   14 MA=NRA
      MB=NRB
      MC=NCB
      GO TO 16
   15 MA=NCA
      MB=NCB
      MC=NRB
   16 DO 23 I=1,MA
      DO 23 K=1,MB
      NC=I+(K-1)*MA
      C(NC)=0.0
      DO 23 L=1,MC
      GO TO (20,21,22),J
   20 NB=L+(K-1)*MC
      NA=I+(L-1)*NRA
      GO TO 23
   21 NB=K+(L-1)*MB
      NA=I+(L-1)*NRA
      GO TO 23
   22 NB=L+(K-1)*MC
      NA=L+(I-1)*NRA
   23 C(NC)=C(NC)+A(NA)*B(NB)
      RETURN
      END
```

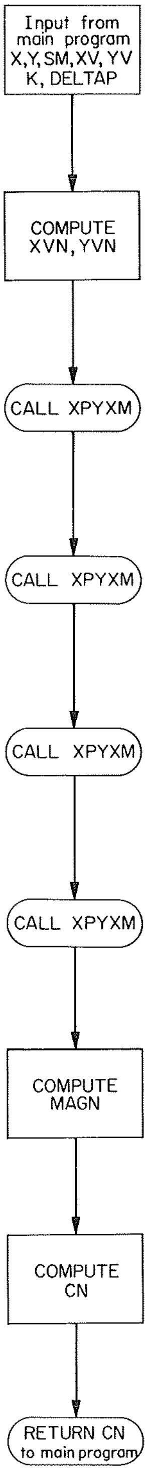
APPENDIX C

FLOW CHARTS

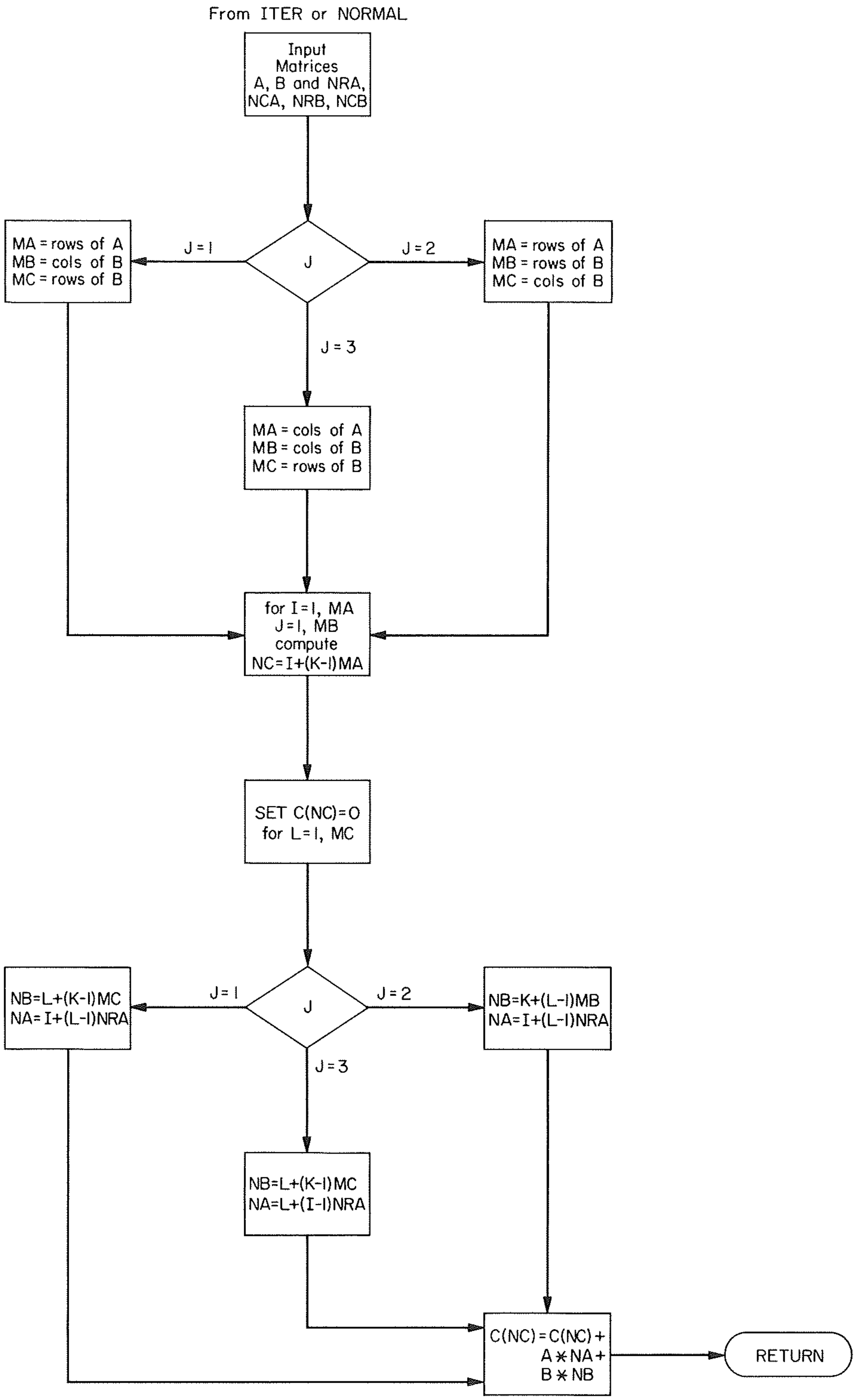
SUBROUTINE REFRC



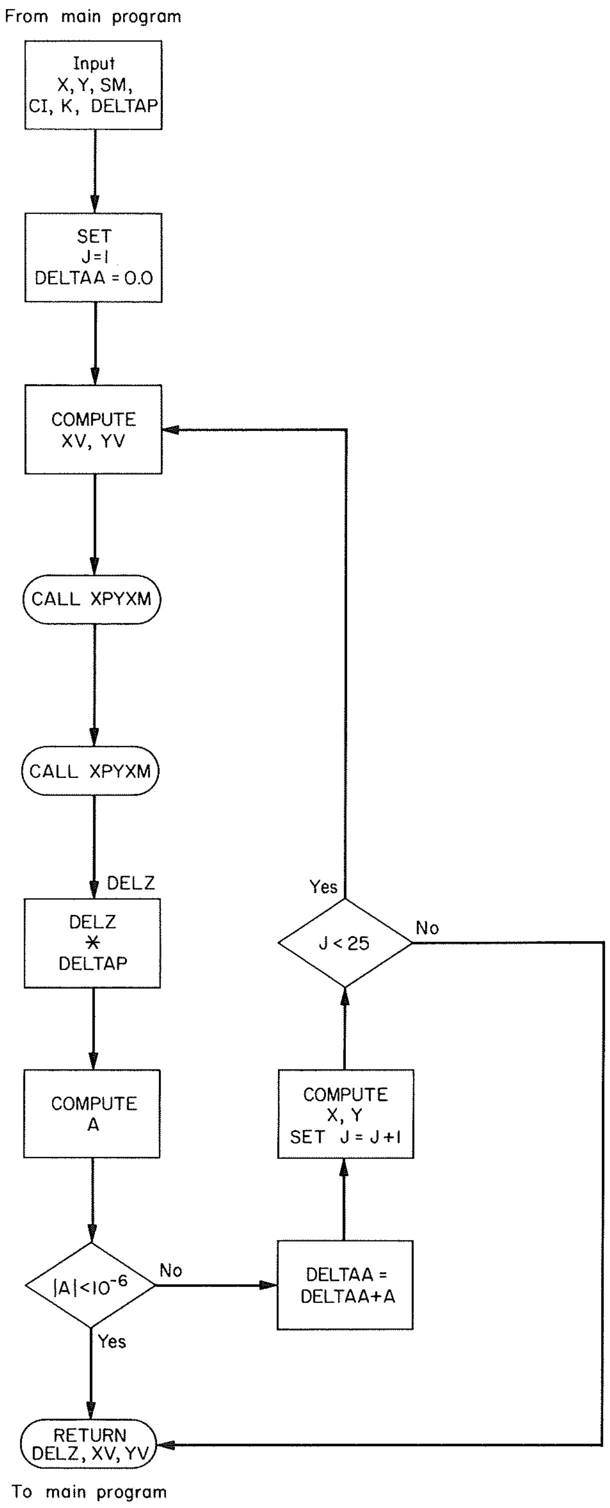
SUBROUTINE NORMAL



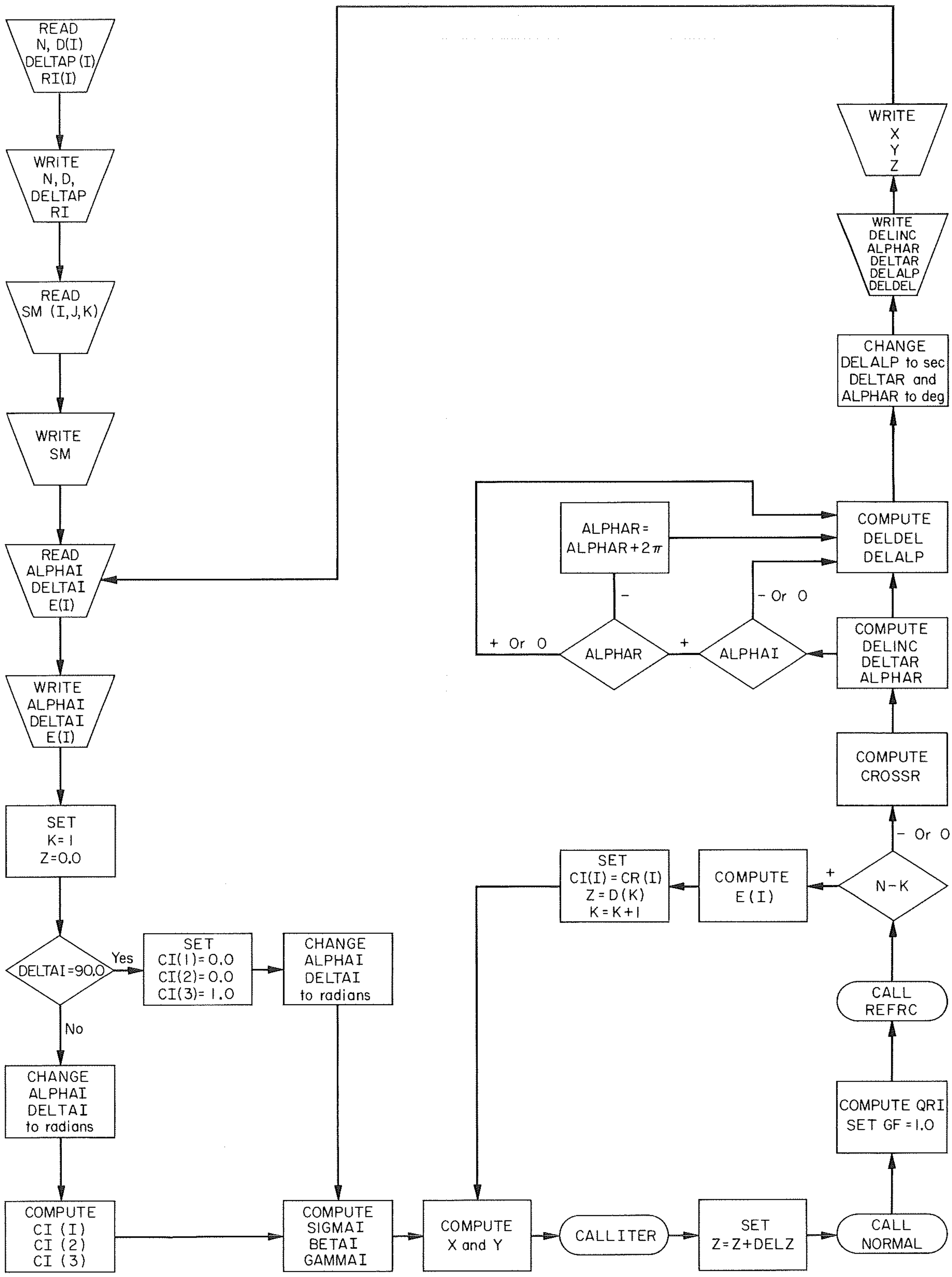
SUBROUTINE XPYXM



SUBROUTINE ITER



MAIN PROGRAM



REFERENCES

1. White, Kenneth C.; and Gadeberg, Burnett S.: Methods for the Prediction of Spacecraft Window-Induced Line-of-Sight Deviations. NASA TN D-5238, 1969.

2. Gadeberg, Burnett S.; and White, Kenneth C.: The Theory of the Correction of Celestial Observations made for Space Navigation or Testing. NASA TN D-5239, 1969.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, D.C. 20546

OFFICIAL BUSINESS

FIRST CLASS MAIL

POSTAGE AND FEES PAID
NATIONAL AERONAUTICS AND
SPACE ADMINISTRATION

POSTMASTER: If Undeliverable (Section 158 Postal Manual) Do Not Return

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Notes, and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION DIVISION

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546